WE CLAIM:

1   1.   A video coder, comprising:

2   a pixel transposition circuit that receives blocks of

3   image data, each block including image data of an array of

4   pixels, said pixel transposition circuit transposing selected

5   pixels on a boundary of a first block with selected boundary

6   pixels of a plurality of blocks neighboring said first block,

7   a   transform   circuit   that   generates   coefficients

8   representative of data of the transposed blocks, and

9   a quantizer that scales the coefficients.

1   2.   The   video   coder   of   claim   1,   wherein   the   pixel

2   transposition circuit transposes alternate pixels along the

3   boundary of said first block with pixels from the neighboring

4   blocks adjacent to the alternate pixels in a transposition

5   direction.

1   3.   The video coder of claim 2 wherein the transposition

2   direction is a diagonal direction, high-left to low-right.

1   4.   The video coder of claim 2 wherein the transposition

2   direction is a diagonal direction, low-left to high-right.

1   5.   The video coder of claim 2, wherein the transposition

2   direction is a vertical direction for pixels along a vertical

3   edge of the first block and a horizontal direction for pixels

4   along a horizontal edge of the first block.

1   6.   A video decoder, comprising:

2   a dequantizer that receives blocks of scaled coefficient

3   information and reconstructing coefficients therefrom,

4   an inverse transform circuit that reconstructs blocks of
5   pixel data from the blocks of reconstructed coefficients, and
6   a pixel transposition circuit that transposes pixels on
7   a boundary of a first block with boundary pixels of a
8   plurality of blocks neighboring said first block and
9   generating blocks of image data for display.

1   7.   The video decoder of claim 6, wherein the pixel
2   transposition circuit transposes alternate pixels along the
3   boundary of the first block with pixels from the neighboring
4   blocks adjacent to the alternate pixels in a transposition
5   direction.

1   8.   The video decoder of claim 7, wherein the transposition
2   direction is a diagonal direction, high-left to low-right.

1   9.   The video decoder of claim 7, wherein the transposition
2   direction is a diagonal direction, low-left to high-right.

1   10.   The video decoder of claim 7, wherein the transposition
2   direction is a vertical direction for pixels along a vertical
3   edge of the first block and a horizontal direction for pixels
4   along a horizontal edge of the first block.

1   11.   A method of encoding image data, comprising the steps
2   of:
3   receiving blocks of image data, each block including
4   data for an array of pixels,
5   transposing selected pixels on a boundary edge of a
6   first block with selected pixels from a plurality of blocks
7   neighboring said first block,

132554 2685:112182

8   transforming pixel data of the blocks from the step
9   transposing to coefficients, and
10   scaling the transformed blocks.

1   12.   The method of claim 11, wherein the transposition step
2   includes transposing alternate pixels along the boundary of
3   the first block with pixels from the neighboring blocks
4   adjacent to the alternate pixels in a transposition
5   direction.

1   13.   The method of claim 12, further comprising a step of
2   generating a transposition keyword representative of the
3   transposition direction.

1   14.   The method of claim 12, wherein the transposition
2   direction is a diagonal direction, high-left to low-right.

1   15.   The method of claim 12, wherein the transposition
2   direction is a diagonal direction, low-left to high-right.

1   16.   The method of claim 12, wherein the transposition
2   direction is a vertical direction for pixels along a vertical
3   edge of the first block and a horizontal direction for pixels
4   along a horizontal edge of the first block.

1   17.   A method of decoding blocks of encoded image data,
2   comprising the steps of:
3   scaling the coded blocks to obtain blocks of coefficient
4   data,
5   transforming the coefficient data of the blocks to pixel
6   data, and

7   transposing selected pixels on a boundary edge of a
8 first block with selected pixels from a plurality of blocks
9 neighboring said first block.

1 18.   The method of claim 17, wherein the transposition step
2 transposes alternate pixels along the boundary of the first
3 block with pixels from the neighboring blocks adjacent to the
4 alternate pixels in a transposition direction.

1 19.   The method of claim 18, further comprising a step of
2 receiving   a   transposition   keyword   that   identifies   a
3 transposition direction.

1 20.   The   method   of   claim   18,   wherein   the   transposition
2 direction is a diagonal direction, high-left to low-right.

1 21.   The   method   of   claim   18,   wherein   the   transposition
2 direction is a diagonal direction, low-left to high-right.

1 22.   The   method   of   claim   18,   wherein   the   transposition
2 direction is a vertical direction for pixels along a vertical
3 edge of the first block and a horizontal direction for pixels
4 along a horizontal edge of the first block.

1 23.   A bitstream generated by a process comprising the steps
2 of:
3   receiving blocks of image data, each block including
4 data for an array of pixels,
5   transposing selected pixels on a boundary edge of a
6 first block with selected pixels from blocks neighboring the
7 first block,

8   transforming pixel data of the transposed blocks to

9 coefficients, and

10   scaling the transformed blocks.

1 24. The method of claim 23, wherein the transposition step

2 includes a step of transposing alternate pixels along the

3 boundary of the first block with pixels from the neighboring

4 blocks adjacent to the alternate pixels in a transposition

5 direction.

1 25. The method of claim 24, further comprising a step of

2 generating a transposition keyword representative of the

3 transposition direction.

1 26. The method of claim 24, wherein the transposition

2 direction is a diagonal direction, high-left to low-right.

1 27. The method of claim 24, wherein the transposition

2 direction is a diagonal direction, low-left to high-right.

1 28. The method of claim 24, wherein the transposition

2 direction is a vertical direction for pixels along a vertical

3 edge of the first block and a horizontal direction for pixels

4 along a horizontal edge of the first block.